

**UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
MIDLAND/ODESSA DIVISION**

VIRTAMOVE, CORP.,

Plaintiff,

v.

AMAZON.COM, INC.;
AMAZON.COM SERVICES LLC; AND
AMAZON WEB SERVICES, INC.,

Defendants.

Case No. 7:24-CV-00030-ADA-DTG

**PLAINTIFF'S RESPONSIVE
CLAIM CONSTRUCTION BRIEF**

TABLE OF CONTENTS

I. Amazon has not shown invalidity of the '814 Patent claims by clear and convincing evidence4

A. “disparate computing environments” (claim 1)4

B. “container” (claim 1).....6

i. Redundant language need not and should not be included.7

ii. “[e]ach container ha[ving] its own execution file” is expressly claimed in dependent claim 2, and should not be imported into independent claim 1.....9

II. Amazon has not shown invalidity of the '058 Patent claims by clear and convincing evidence9

A. “critical system elements” (claim 1)9

B. “functional replicas” (claim 1).....12

C. “shared library” (claim 1)14

D. “forms a part of ... software applications” (claim 1).....16

TABLE OF AUTHORITIES

Cases

<i>Interval Licensing LLC v. AOL, Inc.</i> , 766 F.3d 1364 (Fed. Cir. 2014).....	13
<i>OSRAM GmbH v. Int'l Trade Comm'n</i> , 505 F.3d 1351 (Fed.Cir.2007).....	11
<i>Phillips v. AWH Corp.</i> , 415 F.3d 1303 (Fed. Cir. 2005).....	5, 11
<i>Sonix Tech. Co. v. Publications Int'l, Ltd.</i> , 844 F.3d 1370 (Fed. Cir. 2017).....	12
<i>Thorner v. Sony Computer Ent. Am. LLC</i> , 669 F.3d 1362 (Fed. Cir. 2012).....	6
<i>U.S. Surgical Corp. v. Ethicon, Inc.</i> , 103 F.3d 1554 (Fed. Cir. 1997).....	7

Statutes

35 U.S.C. § 112, ¶ 6.....	14
---------------------------	----

Amazon contends that all asserted claims of both patents asserted in this action are invalid for indefiniteness. Amazon does not and cannot prove invalidity by clear and convincing evidence as required by the law, and its implicit motion for summary judgment should be denied. Amazon also proposes several constructions impermissibly changing the patent scope from the plain and ordinary meaning set out by the applicant and issued by the U.S. Patent and Trademark Office. These constructions should be rejected for the reasons explained below.

I. Amazon has not shown invalidity of the '814 Patent claims by clear and convincing evidence

A. “disparate computing environments” (claim 1)

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
Environments run by standalone or unrelated computers	Indefinite

In the co-pending action *VirtaMove, Corp. v. Google LLC*, Case No. 7:24-cv-00033-ADA-DTG (W.D. Tex.), VirtaMove has proposed construing “disparate computing environments” as “environments run by standalone or unrelated computers.” For consistency across the two actions, VirtaMove proposes the same construction here. This proposal is closely based on the same passage from the specification that Amazon relies on as “lexicography” and that forms the basis of Amazon’s indefiniteness challenge, which the parties also discussed in their pre-filing meet and confer process. *See* Dkt. 71 at 3 (alleging indefiniteness of: “Environments where computers are stand-alone or where there are plural computers and where they are unrelated.”).

Amazon presents two different arguments in favor of indefiniteness. First, that the “unrelatedness” of two computers is indefinite, and second, that the claim *requires* the computers must be “related,” creating a contradiction. As discussed below, these arguments fail because they ignore the fact that a POSITA would understand the scope of “standalone” computers.

To Amazon’s first point, Amazon contends that the claim is indefinite, alleging that “‘unrelated’ is subjective; people may reasonably disagree about whether two things are related or unrelated.” Dkt. 71 at 4. But this entirely ignores the remainder of the claim, which Amazon *admits* does not allow for two computers to be “unrelated” because they must be “part of a single ‘system.’” In other words, Amazon acknowledges that the “unrelated” portion of the specification’s description of “disparate computing environments” cannot fit into the broader context of the claim language. Because the claim *undisputedly* cannot extend to “unrelated” computers, only the “standalone” portion of that description could be relevant to the scope of the claims as a whole. *See Phillips v. AWH Corp.*, 415 F.3d 1303, 1313 (Fed. Cir. 2005) (“Importantly, the person of ordinary skill in the art is deemed to read the claim term ... in the context of the particular claim in which the disputed term appears....”).

Accordingly, the only relevant inquiry (in the context of the claim as a whole) is whether environments run by *standalone* computers is indefinite. Amazon presents no evidence that a POSITA would be unable to understand the boundaries of standalone computers, which is a common phrase used to indicate the ability of computers to operate independently of each other. Amazon’s narrow focus on computers being “unrelated” (a scenario that Amazon acknowledges is simply inapplicable in the context of the asserted claims) ignores whether “standalone” computers can be understood to a POSITA, and Amazon presents no evidence at all that standalone computers would not be understood.

Amazon also alleges in passing that “[n]othing in the patent or prosecution history explains how multiple computers can be part of the same system and yet be stand-alone or unrelated.” Dkt. 71 at 4. As an initial matter, it is *Amazon’s* burden to show indefiniteness by clear and convincing evidence; it is not VirtaMove’s burden to prove definiteness as Amazon implies. And in any event,

two computers can be part of the claimed “system” and still be capable of operating independently, and Amazon presents no evidence to the contrary.

B. “container” (claim 1)

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
No construction necessary; plain and ordinary meaning. In the alternative: An aggregate of files required to successfully execute a set of software applications on a computing platform. Each container for use on a server is mutually exclusive of the other containers, such that read/write files within a container cannot be shared with other containers.	An aggregate of files required to successfully execute a set of software applications on a computing platform is referred to as a container. Each container for use on a server is mutually exclusive of the other containers, such that read/write files within a container cannot be shared with other containers. A container comprises one or more application programs including one or more processes, and associated system files for use in executing the one or more processes; but containers do not comprise a kernel; each container has its own execution file associated therewith for starting one or more applications.

The ’814 Patent specification includes a broad explanation of how a “container” fits within the context of the claimed invention. Although described as a “definition,” in substance the patentee provided an encyclopedia entry, which cannot reasonably be interpreted as pure lexicography, and which would serve only to confuse the jury by substituting a single word in a claim with nearly 100 words of redundant examples of how containers may be implemented. Indeed, Amazon itself omits entire sentences from the supposed “definition” set forth in the specification, confirming that a POSITA would not understand the entirety of its discussion of “container” to be lexicography.

Nor could Amazon have shown that the “exacting” standard for lexicography is met. “To act as its own lexicographer, a patentee must ‘clearly set forth a definition of the disputed claim term’ other than its plain and ordinary meaning.” *Thorner v. Sony Computer Ent. Am. LLC*, 669 F.3d 1362, 1365 (Fed. Cir. 2012). And “[t]he standard for disavowal of claim scope is similarly

exacting. *Id.* The fact that high bar for lexicography is not met here is confirmed not only by the non-definitional nature of the specification’s discussion of a “container,” but also by the fact that two different defendants attempt to apply the alleged “lexicography” in completely different ways.

In particular, Google LLC applies the alleged “lexicography” of the specification to provide a ***substantially different*** proposed “definition” of “container.” *VirtaMove Corp. vs. Google LLC*, Case No. 7:24-cv-00033-DC-DTG, Dkt. No. 63 at 8-10 (W.D. Tex. Oct. 22, 2024). The ***only*** overlap between Amazon’s and Google’s proposed constructions is the first sentence “An aggregate of files...” and the sentence “Each container for use on a server is mutually exclusive....” *Id.* Google does not agree with Amazon that the “execution file” language must be included, except as a fallback position. These disagreements confirm that the entire specification’s explanation of containers need not be part of the construction of “container”

Regardless of how the Court construes “container,” Plaintiff requests that the Court enter identical constructions in both the Google and Amazon actions. Plaintiff believes that the plain and ordinary meaning of “container” applies and is generally consistent with the only two sentences that both Google and Amazon have ***both*** proposed as being definitional: “An aggregate of files...” and “Each container for use on a server is mutually exclusive....” However, as explained below, only “An aggregate of files” would be potentially helpful in a construction for a jury, because the “mutually exclusive...” discussion is redundant with other claim language. This limitation, as well as other limitations Amazon seeks to read into the claim, are addressed below.

i. Redundant language need not and should not be included.

It is black letter law that claim construction “is not an obligatory exercise in redundancy.” *U.S. Surgical Corp. v. Ethicon, Inc.*, 103 F.3d 1554, 1568 (Fed. Cir. 1997). Most of the proposed definition here is redundant of existing claim language. Including two slightly different phrasings

of the same concept would at best confuse the jury, and these redundant sentences should not be included.

“Each container for use on a server is mutually exclusive of the other containers, such that read/write files within a container cannot be shared with other containers.” This sentence is redundant of other express claim language. Claim 1 already recites “a plurality of secure containers of application software” wherein “the application software ***cannot be shared between the plurality of secure containers*** of application software.” Thus, the claim already makes clear which portion of the container is mutually exclusive (i.e., cannot be shared with) other containers, and the “mutually exclusive” portion of Amazon’s construction would inject confusion and surplusage.

“A container comprises one or more application programs including one or more processes, and associated system files for use in executing the one or more processes...” This phrase is redundant with the remainder of Claim 1, which already recites “each container comprising one or more of the executable applications and a set of associated system files required to execute the one or more applications.” Of course application programs include a “process”—otherwise the program would be unable to ***do*** (or “process”) anything. Amazon presents no plausible reason why this language would add anything meaningful beyond the express claim language itself.

“[B]ut containers do not comprise a kernel...” Claim 1 already recites “the containers of application software excluding a kernel.” As with this and the other redundant terms, the only possible purpose for a construction of “container” to include this requirement would be to cause redundancy and confuse the jury.

- ii. “[e]ach container ha[ving] its own execution file” is expressly claimed in dependent claim 2, and should not be imported into independent claim 1.

“[E]ach container has its own execution file associated therewith for starting one or more applications.”

In one non-limiting embodiment, the ’814 Patent specification describes a particular type of configuration data, within a container, that “defines how applications... are started... includ[ing] the definition of a first program to start when a container is installed on a compute platform 40,” as shown in Figure 7. ’814 Patent at 9:8-19 and Fig. 7. As the patent explains, this embodiment “*allows for any number of applications to be started with a container association.*” *Id.* at 9:18-19.

Claim 2 expressly recites this aspect of the preferred embodiment in specifying that “each container has an execution file associated therewith for starting the one or more applications.”

Under Amazon’s proposal, the scope of claims 1 and 2 would be literally coextensive, because the proposed construction requiring a container to have an “execution file associated therewith for starting one or more applications” is identical to the language of claim 2. Amazon also provides no justification for a construction which would, again, be redundant with the express language of claim 2.

II. Amazon has not shown invalidity of the ’058 Patent claims by clear and convincing evidence

A. “critical system elements” (claim 1)

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
Any service or part of a service, “normally” supplied by an operating system, that is critical to the operation of a software application.	Indefinite

Contrasting with Amazon’s other “lexicography” proposals, the ’058 Patent does provide an unambiguous definition of the phrase “critical system element[s],” stating what a CSE is rather than providing examples or embodiments. The definition has two basic requirements: first, the CSE is “‘normally’ supplied by an operating system”; second, it is “critical to the operation of a software application.”

Regarding the former requirement, the patent specification provides further context, explaining: “It is traditionally the task of an operating system to provide mechanisms to safely and effectively control access to shared resources. In some instances the centralized control of elements, critical to software applications, hereafter called critical system elements (CSEs)[,] creates a limitation caused by conflicts for shared resources.” ’058 Patent at 1:22-27. This illustrates the conventional arrangement wherein CSEs are “normally” provided by an operating system (i.e., they are provided by the operating system if the structure of the operating system is not modified beyond its default operation). The specification also provides contrasting examples of the “invention,” consistent with the claims, where “some system elements that are critical to the operation of a software application *are replicated from kernel mode, into user mode....* These system elements are contained in a shared library.” *Id.* at 9:15-19 (emphasis added). The specification parallels the claim requirements and confirms that the OSCSEs recited in limitation 1(b) generally correspond to the operation of a conventional system (where the operating system provides the critical system elements), whereas the SLCSEs of limitation 1(c) generally correspond to a non-conventional aspect of the claimed invention (where the critical system elements are stored in a shared library, outside of the operating system).

Amazon’s indefiniteness challenge stems from the assumption that a POSITA would be unable to determine “whether or not a service was normally found in operating systems.” Dkt. 71

at 8. But Amazon gives no explanation why a survey of other operating systems could plausibly be relevant here. Rather, the relevant determination focuses on the only operating system relevant to the claim scope: the claimed operating system of limitation 1(b).

Second, Amazon’s attack on the word “critical” fails. Amazon poses a series of hypothetical questions divorced from the intrinsic record. Dkt. 71 at 8. References to “crashing” or “perform[ing] unreliably or incorrectly,” with no source other than attorney or expert imagination, do not illuminate the potential claim scope. Neither is Amazon assisted by citation to a dictionary definition that Amazon concedes does not actually define “critical” (but instead defines “criticality,” which is a term that encompasses a “range” of “criticality”) and that Amazon does not even attempt to link to the claim or specification context.

Rather, the Court should look to the text of the patent specification itself, the intrinsic evidence that is the best guide to the patent’s meaning. *See, e.g., Phillips v. AWH Corp.*, 415 F.3d 1303, 1319 (Fed. Cir. 2005) (extrinsic evidence, such as expert reports, “is unlikely to result in a reliable interpretation of patent claim scope unless considered in the context of the intrinsic evidence.”); *OSRAM GmbH v. Int’l Trade Comm’n*, 505 F.3d 1351, 1356 (Fed.Cir.2007) (“The patent specification is the primary resource for determining how an invention would be understood by persons experienced in the field.”).

The ’058 Patent provides numerous examples of critical system elements, more than sufficient to illustrate what elements are “critical.” First, the specification discusses “a TCP/IP stack,” which a POSITA would readily recognize as the core network protocols used for Internet communication. ’058 Patent at 5:41-53. The TCP/IP stack is plainly critical to any application that uses Internet communication. The next examples are additional network services, “including TCP/IP, Bluetooth, ATM; or message passing protocols.” *Id.* at 6:11-13. The specification goes

on to provide specific examples of CSEs that represent extensions or optimizations to file system or network functionalities, such as services to “[a]ccess files that reside in different locations” and network optimizations including “[m]odified protocol processing for custom hardware services.” *Id.* at 6:14-28. In each case, software designed to rely on these services plainly would not function in its intended manner without them.

All of this intrinsic evidence guides a POSITA’s understanding of what services are “critical” and confirms the definiteness of the claim scope. Amazon points to no evidence that any *other* understanding of “critical” would even be considered by a POSITA in the context of the ’058 Patent and the above-cited intrinsic evidence. At the very least, Amazon’s failure even to mention this evidence confirms that Amazon cannot prove indefiniteness by clear and convincing evidence, as required.

B. “functional replicas” (claim 1)

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
No construction necessary; plain and ordinary meaning.	Indefinite

Amazon argues that “replica” has a lexicographic definition, *i.e.* “a CSE having similar attributes to, but not necessarily and preferably not an exact copy of a CSE in the operating system (OS),” and that definition is indefinite as a term of degree because of the word “similar.” Dkt. 71 at 10. This argument fails at both steps.

The Federal Circuit has explained that “[b]ecause language is limited, we have rejected the proposition that claims involving terms of degree are inherently indefinite.” *Sonix Tech. Co. v. Publications Int’l, Ltd.*, 844 F.3d 1370, 1377 (Fed. Cir. 2017). “Thus, a patentee need not define his invention with mathematical precision in order to comply with the definiteness requirement.” *Id.* (internal quotation marks omitted). “Claim language employing terms of degree has long been

found definite where it provided enough certainty to one of skill in the art when read in the context of the invention.” *Interval Licensing LLC v. AOL, Inc.*, 766 F.3d 1364, 1370 (Fed. Cir. 2014). In determining whether the patent has provided sufficient guidance for a term of degree, a reviewing court should “look to the written description for guidance.” *Id.* at 1371.

First, Amazon’s focus on a single sentence from the patent specification ignores the claim context and the full disclosure of the patent specification. In particular, the claim term is “functional replica,” not “replica.” Even if the generic description of “replica” were indefinite (it is not), the limitation to functional replicas provides important clarification.

The specification contains an additional description of the scope of “the term replica” specifically in the context of *functional* replicas: “The CSE library includes replicas or substantial functional equivalents or replacements of kernel functions. The term replica, shall encompass any of these meanings, and although not a preferred embodiment, may even be a copy of a CSE that is part of the OS.” ’058 Patent at 8:27-32; *see also id.* at 9:52-56 (“The term replication means that like services are supplied [*i.e.*, that] essentially a same functionality is provided.”). These sentences explicitly state what scope is “encompass[ed]” by “the term replica”: (1) substantial functional equivalents of kernel functions; (2) replacements of kernel functions; and (3) copies of OSCSEs (*i.e.*, kernel functions). Of these three categories, “substantial functional equivalents” is logically the broadest, since either a replacement or a copy of a kernel function/OSCSE would necessarily also be functionally equivalent.

Accordingly, the phrase “functional replica” does not require mere similarity, but rather (at a minimum) “substantial functional equivalen[ce].” ’058 Patent at 8:27-32. Amazon does not and cannot contend that determining the substantial functional equivalence of two CSEs is indefinite.

Indeed, juries are regularly required to determine functional equivalence in the context of the Doctrine of Equivalents or in the context of 35 U.S.C. § 112, ¶ 6.

C. “shared library” (claim 1)

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
No construction necessary; plain and ordinary meaning. In the alternative: An application library <i>whose</i> code space <i>is</i> shared among all user mode applications.	An application library code space shared among all user mode applications.

The term “shared library” appears throughout the specification and claims of the ’058 Patent. It has a plain and ordinary meaning that is confirmed by the claim context and by the specification. Instead of recognizing this plain and ordinary meaning, Amazon demands including a confusing, circular “definition” that introduces additional terms unfamiliar to a jury and that, appears to contradict the claim context.

For example, the patent specification makes clear that “code space” refers to where a library is located, not to the library itself. *See, e.g.*, ’058 Patent at 3:39-45 (“the same set of instructions in the same physical memory space, *that is, shared code space...*”); *id.* at 6:54-55 (“Static library: An application *whose* code space is *contained* in a single application”); *id.* at 7:3-5 (“[W]hat is commonly done is to provide an application library *in* shared code space, which multiple applications can access.”). This usage, which reflects the plain and ordinary meaning of “code space” to a POSITA, contradicts the notion that a shared library is *defined* as “an application library code space” as Amazon requests.

There is a simple explanation for the confusing construction, though: the patent applicant obviously introduced a pair of typographical errors into the definition of “Shared library.” The original version of this definition, in the provisional application to which the ’058 Patent claims priority, is shorter: “An application library *whose* code space *is* shared among all user mode

applications.” Ex. 1 (Provisional Patent Application No. 60/504,213) at 9. That definition cleanly flowed from the definition of “Application library” above it, and paralleled the definition of “Static library” below it, confirming that the key difference between a shared library and a static library is whether the code space is contained in a single application or shared among applications:

Application library: A collection of functions in an archive format that is combined with an application to export system elements.

Shared library: An application library whose code space is shared among all user mode applications.

Static library: An application library whose code space is contained in a single application.

Id.

When the applicant revised the provisional specification to form the non-provisional application, additional detail was added to the definition, but the words “whose” and “is” were removed. Those words were not deleted from the definition of “Static library,” which retains the same definition in the final specification. The new language includes “The code space is different than that occupied by the kernel,” confirming that “code space” is a space occupied by code, not code itself. This confirms that the deletion of “whose” was unintentional, and that the correct interpretation should retain the original language of the provisional. A POSITA reading the specification would readily understand that this is the correct interpretation. Therefore, if the extent the Court believes construction is necessary, the correct definition without the typographical errors should be included: “An application library *whose* code space *is* shared among all user mode applications.”

D. “forms a part of ... software applications” (claim 1)

Plaintiff’s Proposed Construction	Defendants’ Proposed Construction
No construction necessary; plain and ordinary meaning.	resides in the same address space as application code

Amazon’s only asserted basis for this construction is prosecution disclaimer. Dkt. 71 at 12-13. The applicant distinguished prior art on the basis that the prior art’s disclosure of a “proxy” was not an SLCSE because it did not “reside in the same address space as application code.” In context, this simply states that a logical prerequisite for the “forms a part” limitation is not met; something cannot form a part of the software application if it does not even share an address space with that software application. The applicant’s statement is not definitional, *i.e.* does not state or logically imply that “resides in the same address space as application code” is coextensive with the “forms a part” limitation. Amazon’s proposal risks reading “forms a part” out of the claim entirely, and should be rejected.

Dated: November 12, 2024

Respectfully submitted,

/s/ Reza Mirzaie

Reza Mirzaie
CA State Bar No. 246953
Marc A. Fenster
CA State Bar No. 181067
Neil A. Rubin
CA State Bar No. 250761
Amy E. Hayden
CA State Bar No. 287026
Jacob R. Buczko
CA State Bar No. 269408
James S. Tsuei
CA State Bar No. 285530
James A. Milkey
CA State Bar No. 281283
Christian W. Conkle
CA State Bar No. 306374
Jonathan Ma

CA State Bar No. 312773
Daniel Kolko
CA State Bar No. 341680
RUSS AUGUST & KABAT
12424 Wilshire Boulevard, 12th Floor
Los Angeles, CA 90025
Telephone: 310-826-7474
Email: rmirzaie@raklaw.com
Email: mfenster@raklaw.com
Email: nrubin@raklaw.com
Email: ahayden@raklaw.com
Email: jbuczko@raklaw.com
Email: jtsuei@raklaw.com
Email: jmilkey@raklaw.com
Email: cconkle@raklaw.com
Email: jma@raklaw.com
Email: dkolko@raklaw.com

Qi (Peter) Tong
4925 Greenville Ave., Suite 200
Dallas, TX 75206
Email: ptong@raklaw.com

**ATTORNEYS FOR PLAINTIFF
VIRTAMOVE, CORP.**

CERTIFICATE OF SERVICE

I certify that this document is being served upon counsel of record for Defendants on November 12, 2024 via electronic service.

/s/ Christian W. Conkle